**Python osnove**

**a)** Implementirati iduće funkcije:

* Napisati rekurzivnu count funkciju. Funkcija prima listu i predikat i vraća koliko elemenata u listi zadovoljava predikat. Predikat je funkcija koja prima jedan element liste i vraća True / False.
* Napisati rekurzivnu funkciju koja generira listu stringova koji predstavljaju sve moguće kombinacija slova „A“, „B“ i „C“ neke zadane dužine. Dužina stringova je zadana kao parametar funkcije. Funkcija će imati dodatni parametar za prosljeđivanje djelomičnih stringova.
* Napisati iterativnu funkciju koja generira listu stringova koji predstavljaju sve moguće kombinacija slova „A“, „B“ i „C“ neke zadane dužine. Dužina stringova je zadana kao parametar funkcije. Funkcija koristi stog umjesto rekurzije.
* Napisati funkciju koja prima listu pozitivnih cijelih brojeva. Funkcija ispisuje sve kombinacije brojeva u listi koju zadovoljavaju iduću jednadžbu, „brute-force“ algoritmom:

(zbroj brojeva)2 % 23 = 0

Funkcija će imati dodatni parametar za prosljeđivanje trenutne liste brojeva u zbroju.

**b)** Implementirati dvije klase (Point2D i Polar2D) koje predstavljaju dvodimenzionalnu točku u prostoru kao par kartezijevih i par polarnih koordinata. Obje klase će imati iduće funkcionalnosti koje će :

* Inicijalizaciju sa dvije koordinate za Point2D ili (kut, magnituda) za Polar2D
* Pretvaranje u/iz polarnih koordinata u kartezijeve. Metoda vraća odgovarajuću klasu.
* Metodu \_\_repr\_\_ za pretvaranje u string
* Negaciju koordinate (unarni operator -) odnosno vektora. Metoda vraća novi objekt iste klase.
* Zbroj dvije koordinate (binarni operator +) kao da su vektori. Metoda vraća novi objekt iste klase.
* Euklidova udaljenost među koordinatama
* Usporedbu == i !=

Formule za pretvaranje u polarne koordinate:

kut, magnituda = atan2(y, x), (x\*\*2+y\*\*2) \*\* 0.5

Formule za pretvaranje u kartezijeve koordinate:

x, y = cos(kut)\*magnituda, sin(kut)\*magnituda

Sve metode i operatori rade podjednako za obje klase u bilo kojoj kombinaciji. Za usporedbe jednakosti koristiti abs(a-b) < 0.01 umjesto a == b zbog ograničene preciznosti decimalnih brojeva.

Funkcije i klase testirati na priloženom kôdu.